

Linux Update

Jon Titus, Senior Technical Editor



A view of Linux from several perspectives will help embedded-system designers better understand how they can use this open-source operating system. Experts at Eurotech, Texas Instruments, and Rowebots share their approaches.

"Contrary to what some engineers might think, Linux provides a mature operating system," said Arlen Nipper, president and CTO at Eurotech. "You can obtain best-in-class security, TCP/IP stacks, and support for wireless networking, for example. The associated code drops into Linux and works right away."

"Customers can get everything they need; cross-compiler tools ready to run a development system, all the linux source code, utilities to run Python, embedded Java, or XWindows, and so on," noted Nipper. "So engineers who use Linux get a complete software-development system and run-time code at essentially no cost."

Talk with the IT Department



According to Nipper, IT departments will play a larger role in the development of embedded systems because most new systems communicate with computers overseen by IT people and with programs used by corporate managers. "Programmers might write stand-alone code for a new packaging controller," explained Nipper. "But then the IT

Linux Update

Published on Electronic Component News (<http://www.ecnmag.com>)

people say, 'We need to know how many packages went through a machine during time X.' That requirement can become a nightmare unless the programmers combined, say, embedded Linux and Java within a flexible framework. That approach lets them easily make changes."

Some aspects of Linux use require careful attention. "There are patent issues with MP3 code, for example, so I suggest developers use the open-source Ogg Vorbis audio player," said Nipper. "Developers also must understand what the GPL [general public license] covers and what it doesn't. There's still a misconception that if an application runs on Linux, its code becomes open source. That's not true. But if you modify an open-source Linux driver, it does become part of the Linux-community's code."

Focus on your Application, Not on Linux

"Engineers look to Linux as projects become more complex," said Jason Kridner, open platforms principal architect at Texas Instruments. "They now need networking stacks, support for many types of storage devices, USB ports and devices, and so on. Linux has a rich set of tools, code, and community support that people can use as a starting point. Then they can focus on what they want their device to do."

Although Linux can simplify development, it can challenge newcomers who jump in unprepared. "If you have used Linux on a PC, getting up to speed is not as great a challenge," noted Kridner. "The biggest difference between Linux on a small Beagle Board and on a PC centers on booting up the kernel." The Beagle Board project, supported by its own community or user/developers, offers a USB-powered board that uses a TI OMAP3530 processor with an integrated ARM Cortex-A8 CPU and additional multimedia accelerators. "The board offers several boot options that include booting directly from an SD Card, which simplifies start up," explained Kridner. "Once you get beyond the boot-up steps and run Linux, it behaves the same way as Linux on a PC, except it consumes a lot less power."

"The GPL (ver 2) covers the Linux kernel, and you may have other licenses to manage code you want to add to an application," added Kridner. "But a large portion of the applications that run on Linux are available under open source licenses."

Overcome GPL Concerns

Application programs on most Linux-based systems don't incorporate kernel code within the application itself. This clean separation keeps proprietary code from entering the open-source universe. If you want to use Linux in a microcontroller, though, you may need to link a Linux kernel into your application to create a single compact block of code. Then code does become open source. You can bypass this aspect of the Linux GPL, though.

Kim Rowe, the founder of Rowebots, crafted a POSIX-compatible OS that looks and operates like Linux, but that does not include any code protected by a GPL. POSIX stands for Portable Operating System Interface standard, IEEE 1003.1. In fact, Rowe's work predates Linux. "We built a completely 100-percent-compatible Linux

Linux Update

Published on Electronic Component News (<http://www.ecnmag.com>)

that keeps developers away from the GPL," said Rowe. "Our implementation looks exactly the same, works exactly the same, and passes all the same test suites as Linux. This represents a new concept."

"We offer two types of standardization, continued Rowe. "First, we meet the POSIX standard for compatibility. Second, we offer Linux compatibility, which means you can take a Linux application and without making changes, run it directly on our operating system. So you have another level of standardization over and above POSIX, which means you create portable code." Rowebots offers a 16-bit (DSPnano) and a 32-bit version (Unison) of its OS. Both offer the same APIs and functions. "We have created a 1-Kbyte version for Coldfire processors and we run our OS on microcontrollers such as the Renesas M16 and Microchip PIC24."

Rowebots offers a free development license for noncommercial use. "If you want a commercial license, you purchase it, but you can make that decision after you finish development work," said Rowe.

For further reading

For more information about the Beagle Board Project, visit:

http://jkridner.s3.amazonaws.com/esc/BeagleBoard101_20081025.ppt [1].

Source URL (retrieved on 04/25/2015 - 7:03pm):

<http://www.ecnmag.com/articles/2009/01/linux-update>

Links:

[1] http://jkridner.s3.amazonaws.com/esc/BeagleBoard101_20081025.ppt